보안 라우터를 위한 보안 감사추적 시스템

Security Audit System for Secure Router

두소영(S.Y. Doo) 보안운영체제연구팀 선임연구원
김정녀(J.N. Kim) 보안운영체제연구팀 책임연구원, 팀장
장종수(J.S. Jang) 네트워크보안그룹 책임연구원, 그룹장

감사추적기능은 시스템의 공격을 방어하기 위한 마지막 방법 중 하나이다. 공격을 사전에 차단하거나 공격이 이루어지는 순간에 실시간으로 대응하는 능동적 방법이라기보다 공격이 이루어진 후에 시스템에 남겨진 정보를 분석하여 공격의 유형과 상황을 유추해내기 위한 수동적인 방법이다. 본 논문에서는 감사추적 기능이 보안 네트워크 장비에서 중요한 이유를 설명하고 반드시 기록으로 남겨야 하는 내용이 어떤 것인지 정리하며 실제로 보안 라우터에 구현된 보안 감사추적 시스템의 구성과 기능을 설명한다. 본 논문에서는 일반 감사기록과 보안 감사기록을 분리하여 관리하고 있는데 그 필요성에 대해 설명한다.

I. 서 론

감사추적 기능은 시스템의 공격이 이루어진 후에 공격의 내용과 방법을 분석할 수 있는 중요한 기능이다. 감사추적 기능 중 가장 일반적인 것으로는 유닉스 계열 시스템에서 사용하는 syslog를 예로 들 수 있다. 네트워크 장비에 탑재되어 있는 기본 운영체제 시스템들은 이 syslog를 이용하여 운용중에 발생하는 중요한 사건을 기록하고 있다. 최근 네트워크 장비와 보안 장비들이 하나의 시스템으로 융화되어 보안 네트워크 장비로 탄생하고 있다. 보안 스위치, 보안 라우터, 그리고 보안 게이트웨이는 장비 고유 기능에 내부 또는 외부 망으로 패킷을 전달하기 전에 유해 여부를 판단하기 위한 침입탐지 기능 및 침입차단 기능을 추가한 형태의 장비이다. 이렇게 일반적인 네트워크 장비 기능 외에 보안이라는 복합적인 기능을 가지고 있는 네트워크 장비들은 망과 장비 자체에 기록되어야 할 의미 있는 사건들이 다양하게 대량 발생된다. 이러한 사건을 syslog를 이용하여 기록하는 것은 불가능한 것은 아니지만 남겨진 로그들을 분석하고 활용하기에 효율적이지 못하다. 또한, 보안에 관련된 중요한 감사기록은 다른 감사기록 내용과 달리 삭제되거나 수정되지 않을 수 있는 보호방법도 마련되어야 한다.

본 논문에서는 이러한 한계를 해결하기 위해서 보안 감사추적 기능을 이용하여 체계적이고 안전한 보안 감사기능을 설계하고 구현하였다.

본 논문의 구성은 먼저, Ⅱ장에서는 시스템의 로그 생성과 분석에 관한 연구들을 살펴보고, Ⅲ장에서는 보안 감사기록에 관한 요구사항을 정리하고, Ⅳ장에서는 본 논문에서 제안하고 있는 보안 감사추적 시스템을 보안 라우터에 구현한 내용을 설명한 후 Ⅴ장에서 결론을 맺고자 한다.

Ⅱ. 관련연구

로그파일을 이용한 공격 탐지 및 분석 방법은 새로운 것이 아니며 Swatch[1],[2]와 TkLogger[3]와 같은 가장 단순한 형태의 경우 로그 파일의 내용을 정형화된 표현과 비교하여 특정 사건을 찾아내는 것이 일반적으로 사용되어 왔다. 좀더 높은 수준의 툴인 ASAX(Advanced Security audit trail Ana-

lyzer on uniX)[4]는 RUSSEL이라고 불리는 규칙 기반의 언어를 이용하여 감사추적을 처리하고 TCSEC[5] C2 등급을 만족하도록 설계되어 있다. 다양한 보안 툴에 의해서 생성되는 로그와 리포트를 분석하기 위해서 제안된 SAINT(A Security Analysis Integration Tool)[6]도 있다. 이 툴에서는 시스템에 문제를 발생시키는 어떤 패턴을 찾기 위한 시도와 이에 대한 경고 메시지를 생성하고 가능한 해결책도 제시하려고 하고 있으나 그 목표를 달성하기 위해서는 아직 해결해야 할 문제들이 남아 있다.

로그 파일을 해커로부터 보호하기 위한 방법[7]도 연구되고 있으며 복잡하고 다양하게 발생된 로그를 쉽게 보여주기[8] 위한 연구도 행해지고 있다.

자체 개발한 보안 시스템의 로그를 관리하고 분석하기 위한 것으로 NIDES(Next generation Intrusion Detection Export System)도 이러한 감사추적 기능을 포함하고 있다[9]-[11]. NIDES의 감사추적 로그 내용은 사용자들의 시스템 운용 패턴 분석에 중점을 두고 있다.

SRS(Secure Router System)는 보안 라우터 시스템으로 PC 기반의 리눅스 라우터에 침입탐지, 패킷필터링, 신뢰채널 기능을 추가하여 설계한 것이다[12]-[16]. 이 시스템은 보안 망을 구성할 때 IDS(Intrusion Detection System), 방화벽 등 별도의 시스템을 이용하여 내부 망을 보호하던 구성을 간단한 형태로 대체할 수 있으며 분리되어 처리되던 기능을 서로 유기적으로 연동시킬 수 있다는 장점을 가지고 있다.

이 논문에서는 보안 라우터 시스템에서 발생된 보안에 관련된 감사 정보를 별도로 분리하여 관리하는 감사추적 시스템을 제안한다. 보안 감사 기록을 별도로 수집하고 관리하여 기존 시스템 로그와 분리시키고 있는데 그 이유는 보안 기능의 로그를 종합적으로 관리 및 분석할 수 있고 발생된 로그를 보다 안전하게 보호하기에 편리하기 때문이다.

Ⅲ. 보안 감사 요구사항

보안 감사기록은 남겨진 로그를 분석하는 것도 중요하지만 기본적으로 어떤 내용을 어떻게 기록하느냐가 더 중요하다. 따라서 제공되는 감사추적 인터페이스를 이용하여 정확한 사건의 내용을 단계별로 구분하여 기록하는 것이 필요하다.

본 연구팀에서 개발된 보안 라우터 시스템은 패킷필터링 기능, 침입탐지 기능, 시스템자체 보안 기능, 신뢰채널기능을 가지고 있다.

본 연구팀에서 개발된 보안 라우터 시스템을 고려하여 본 논문에서는 다음과 같은 사건들을 주요 사건으로 분류하고 각각의 사건에 대한 중요 정보들을 정리해 보았다.

패킷필터링 기능은 라우터에 유입되는 모든 패킷의 주소, 포트, 프로토콜을 검사하여 그들의 수신처로 전달할 것인가를 판단하는 것이다. 이 기능에서 중요 사건으로는 패킷필터링을 위한 규칙의 설정, 변경 및 삭제 사건과 권한이 없는 사용자가 규칙을 조작한 사건, 그리고 규칙에 위배되는 패킷의 입력과 그 처리 결과가 있다.

침입탐지 기능은 라우터에 유입된 모든 패킷의 헤더와 패킷의 내용까지 분석하여 특정 공격을 찾아내는 것이다. 이 기능에서 중요 사건으로는 정책 설정, 변경 및 삭제, 권한이 없는 사용자가 정책을 설정하고자 시도한 것, 정책에 의해서 공격으로 판단된 경우 등이 있다.

시스템 자체 보안 기능은 라우터 자체를 공격하는 것을 방어하기 위한 것으로 시스템 관리자가 될 경우 시스템 자원과 정보가 위험에 처하는 것을 막기 위한 것이다. 일반적으로 비밀번호나 인증서만을 사용하고 있으나 그것만으로는 부족하다고 판단되어 보안 라우터에서는 역할기반 접근제어[17] 기능을 커널 레벨에서 제공하도록 하고 있다. 이 기능에서 중요한 사건으로는 라우터의 시동 및 종료, 라우터의 모듈화되어 있는 모든 기능별 시작과 종료, 라우터에 접근하는 관리자에 대한 식별 및 인증, 원격으로 접근하는 경우 식별 및 상호 인증, 라우터 자원과 정보에 대한 접근통제 규칙 설정, 변경 및 삭제, 라우터에서 제공되는 통신 프로토콜들의 시작 및 종료, 라우터에 접근할 수 있는 사용자 신원확인 관련 데이터의 설정, 변경 및 삭제 등이 있다.

신뢰채널 기능은 라우터에 연결되어 있는 단말간에 전달되는 모든 데이터를 안전하게 전달할 수 있는 가상 사설 망을 제공하는 것이다. 이 기능에서 주요한 사건으로는 신뢰채널 설정 및 삭제, 변경, 권한이 없는 사용자의 신뢰채널 설정 및 삭제, 변경 시도, 전송 데이터 무결성을 위반한 통신 프로토콜, 설정된 신뢰채널의 비정상적인 종료 또는 재 연결 등이 있다.

그 외에도 보안 라우터에서는 감사추적기록을 위해 각 단계를 설정 및 삭제, 변경하는 사건과 권한이 없는 사용자의 이러한 동작에 대한 기록도 필요하다.

보안 감사기록 레코드에 저장되는 정보는 주체 및 객체의 식별자, 사건 발생 날짜와 시간, 사건의 형태, 사건의 성공 또는 실패 여부, 실패한 사건의 경우 실패의 이유 등이 필요하다.

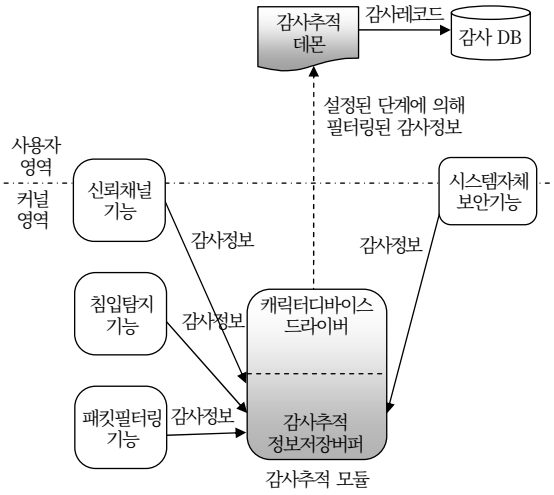
이 외에 보안 감사기록에 고려되어야 하는 내용으로는 감사대상 사건의 사전/후 선택, 데이터 압축, 다중의 감사추적, 물리적 저장 매체 등을 생각할 수 있다. 이 논문에서 감사 기록은 사건이 발생되는 사후 기록되며 데이터는 압축하지 않고 파일 크기가 일정 한계를 넘어서는 경우 다른 파일을 생성하여 저장하도록 하였다. 감사추적 데이터는 보안을 위해서 특정한 역할이 있는 관리자만이 감사 단계를 설정할 수 있고 감사기록에 접근이 가능하도록 역할기반 접근제어를 이용하여 보호하고 있다.

Ⅳ. 보안 감사추적 시스템

보안 사건을 기록하는 감사추적 시스템은 감사정보를 모으는 감사추적 모듈과 사건 정보를 적절한 형태로 만들어서 파일에 저장하는 감사추적 데몬으로 구성된다. 감사추적 모듈은 리눅스 모듈 형태로 개발되어 필요한 경우 올리고 불필요한 경우 제거할 수 있는 편의성을 갖추고 있다.

(그림 1)에 감사추적 시스템과 다른 기능들과의 연관성을 나타내고 있다.

감사추적 데몬은 응용 계층에서 수행되고 감사추적 모듈과는 캐릭터 디바이스를 통해서 데이터를 주고 받는다. 저장된 감사추적 파일을 보기 위한 뷰어



(그림 1) 감사추적 시스템 구성도

가 GUI 형태로 제공된다[18].

감사추적을 위한 로그는 각 기능별로 분리되고 각 기능에 따라 서로 다른 단계를 설정할 수 있다. 보안 라우터에 있는 기능에 따라서 시스템 자체 보안기능, 패킷필터링 기능, 침입탐지 기능, 신뢰채널 기능, 감사추적 기능에 대해 설정할 수 있고 설정 단계는 시스템을 사용할 수 없을 정도의 비상사태(SEC_EMERG), 즉시 조치가 필요한 상태(SEC_ALERT), 위급한 상태(SEC_CRIT), 일반적인 에러 상태(SEC_ERR), 경고 상태(SEC_WARNING), 정상적인 상태이지만 중대한 상황(SEC_NOTICE), 정보 메시지(SEC_INFO), 디버그 메시지(SEC_DEBUG) 8단계로 나뉘어지고 비상사태가 가장 심각한 상태이고 정보 메시지가 가장 낮은 단계이다. 낮은 단계를 선택한 경우 상위 단계를 포함하여 로그가 남겨진다.

개발된 감사추적 시스템은 두 가지 경로를 통해서 로그를 발생시킬 수 있다. 하나는 커널 내부 함수들을 위한 내부 함수 호출을 통한 로그 발생 경로이고 다른 하나는 응용 계층에서 발생되는 로그들을 기록하기 위한 시스템 호출을 통해서 발생된다.

보안 라우터에서는 각각의 기능을 모듈화하여 구현한 것이 특징이다. 모듈간의 정보 전달을 위해서 특정함수를 이용할 경우 사용되는 방법을 이용하여 로그정보를 전달할 수 있도록 하였다.

커널 내부 함수들의 중요 사건을 기록하기 위한 방법은 (그림 2)에 간략한 형태로 나타내었다. register_aud(int (*func)), mod_aud_get_func( ) 함수를 커널 내부에 정의해 두고 이 함수를 export 시킨 후 감사추적 모듈의 초기화 단계에서 func 함수를 local_audit_aud_write_sys에 연결하는 형태이다. 응용 프로그램에서 발생되는 감사추적 로그를 위해서는 int sys_aud_write_sys(int block_id, int aud_level, char * fmt, ...)를 사용하여 기록할 수 있다.

본 논문에서는 보안 감사뿐 아니라 사용자 또는 관리자가 시스템 호출을 수행한 경우에 대해서도 모

```
/* auditmodule.c */

int init_module(void)
{
        .........
        register_aud(local_audit_aud_write_sys);
        .........
}
int local_audit_aud_write_sys
(int block_id, int aud_level, char *log, ...)
{
         ......................
}

/* audit_kernel.c */

int (*aud_get_func)(int block_id, int aud_level, char *log, ...);

int register_aud
(int (*func)(int block_id, int aud_level, char *log, ...))
{

    aud_get_func = func;
    if(aud_get_func == NULL){
       printk("register_aud : auditmodule not yet loaded\n");
       return 1;
    }else {
       printk("register_aud : auditmodule loaded\n");
       return 0;
    }
}
int mod_aud_get_func(int block_id, int aud_level, char *fmt, ...)
{
    ......................
    returncode = (*aud_get_func)(block_id, aud_level, "%s", fmt);
    if(returncode == 0){
       return 0;
    }else if(returncode == 1){
       return 0;
    }else {
       printk("mod_aud_get_func : aud_get_func called error = %d \n",
                 returncode);
        return returncode;
    }
}

EXPORT - SYMBOL(register_aud);
EXPORT_SYMBOL(mod_aud_get_func);
```

(그림 2) 보안 감사추적 처리를 위한 함수들

든 로그를 생성한다. 시스템 호출이 발생된 경우 감사추적 시스템에서 이를 가로채기 하여 감사정보 레코드를 생성하고 이를 저장하도록 되어 있다. 또한, 보안 라우터에서는 모든 시스템 호출에 대해 시스템 자체 보안을 위한 역할 기반 접근제어에 의해서 권한이 없는 경우 자원과 정보에 접근하는 것을 통제하는 기능도 제공하고 있다. 감사추적 모듈에서는 시스템 호출을 가로채기 하여 감사레코드를 생성한 후 접근제어 규칙에 위배되지 않으면 시스템 호출을 정상적으로 처리하도록 하고 그렇지 않은 경우에는 시스템 호출을 오류메시지와 함께 중지시킨다. 일반적인 시스템 호출의 경우 setuid와 같은 시스템 호출은 해킹 유형을 찾아내는 중요한 요소가 되기 때문에 보안 감사 기록에 저장된다.

개발된 감사추적 시스템에서 사용되는 로그 정보의 내용은 아래 예제와 같이 (호스트이름, 사건이름, 사건발생시각, 사건주체정보, 응용프로그램정보, 감사단계, 감사정보, 결과, 발생순번)을 포함하고 있다.

```
sos62 SRSAudit event,
aud_write_sys( ),
Wed Oct 13 11:33:10 2004
user, root(0), root(0), root(0), root(0)
process, 1916, aud_select
block_id:ATEB
aud_level:SEC_NOTICE
info:,aud_select system call
return, 0
sequence, 1
```

V. 결 론

본 논문에서는 보안 네트워크 장비의 보안 감사 기록을 위한 요구사항을 살펴보고 보안 라우터를 예로 들어 감사 기록 대상과 내용을 살펴보았다. 본 논문에서는 TCSEC B2 등급을 만족하는 감사기록 내용을 목표로 그 예를 설명하고, 보안 감사기록을 별도로 남기도록 하여 효율성을 고려하고 있다. 현재까지는 로그의 질을 높이기 위해서 로그를 발생시키

기 위한 다양하고 편리한 인터페이스에 연구 초점을 맞추었다. 추후에는 남겨진 로그를 자동으로 분석하고 효과적으로 처리할 수 있는 기능을 확장 개발하고자 한다.

참 고 문 헌

[1] Stephen E. Hansen and E. Todd Atkins. "Centralized System Monitoring with Swatch," *UNIX Security Symposium*, Sep. 1992, pp.105-117.

[2] Stephen E. Hansen and E. Todd Atkins. "Automated System Monitoring and Notification with Swatch," *LISA*, 1993, pp.145-155.

[3] Doug Hughes. TkLogger. Program available at "ftp://coast.cs.purdue.edu/pug/tools/unix/tklogger.tar.Z."

[4] N. Haller, B. Charlier, A. Mounji, and I. Mathieu, "ASAX: Software Architecture and Rule-based Language for Universal Audit Trail Analysis," *ESORICS*, Nov. 1992, pp.435-450.

[5] S. Chokhani, "Trusted Products Evaluation, Process" *Communications of the ACM*, Vol.35, No.7, July 1992, pp.64-76.

[6] D. Zamboni, "SAINT: A Security Analysis Integration Tool," *In Systems Administration, Networking and Security Conference*, May 1996, pp.3-15.

[7] M. Bellare and B. Yee, *Forward Integrity for Secure Audit Logs*, Tech. Rep., Computer Science and Engineering Department, University of California at San Diego, Nov. 1997.

[8] J. Hoagland, C. Wee, and K.N. Levitt, "Audit Log Analysis Using the Visual Audit Browser Toolkit," *U.C. Davis Computer Science Department Technical Report CSE-95-11*, 1995.

[9] T. Lunt, "Automated Audit Trail Analysis and Intrusion Detection: A Survey," *National Computer Security Conf.,* Oct. 1988.

[10] R. Jagannathan, T.F. Lunt, F. Gilham, A. TRamaru, C. Jalali, P. Neumann, T.D. Garvey, and J. Lowrance, *Requirements Specifications: Next Generation Intrusion-Detection Export System(NIDES)*, Technical Report, SRI International, Sep. 1992.

[11] T.F. Lunt, "Detecting Intruders in Computer Systems," *Conference on Auditing and Computer Technology*, 1993.

[12] B.H. Jeong, J.N. Kim, S.W. Son, and C.H. Park, "Krnel-level Intrusion Detection System for Minimum Packet Loss," *ICACT*, Feb. 2004, pp.207-212.

[13] 정보흥, 김정녀, 장종수, "라우터 보안 기술 동향," 주간기술동향 , 통권 1154호, 2004. 7., pp.1-11.

[14] 한민호, 김정녀, 손승원, "VPN에서 보안 메커니즘," NCS, Dec. 2003, pp.97-100.

[15] 임재덕, 김정녀, "공개된 인터넷 키 교환 프로토콜 구현에 대한 규격 시험," 통신학회 하계종합학술발표회, July 2004, p.166.

[16] 두소영, 김정녀, "네트워크 장비를 위한 자체 보안 시스템," *CEIC*, Dec. 2004, pp.219-222.

[17] R. Sandhu, D. Ferraiolo, and R. Kuhn. "The NIST Model for Role Based Access Control: Towards a Unified Standard," *Proc., 5th ACM Workshop on Role-Based Access Control*, 2000.

[18] 조수형, 은성경, 김정녀, "라우터의 안전한 네트워킹을 위한 정책 기반의 보안 관리," 한국통신학회 추계종합학술대회 논문집, Nov. 2004, p.175.